Categorical Data Analysis Exercise Solutions

Set-up

```
library(tidyverse) # for tidyverse
library(here) # for file paths
library(survey) # for survey analysis
library(srvyr) # for tidy survey analysis

anes <- read_rds(here("Data", "anes_2020.rds")) |>
  mutate(Weight=Weight/sum(Weight)*231592693)
# adjust weight to sum to citizen pop, 18+ in Nov 2020 per ANES methodology documentation
anes_des <- anes |>
  as_survey_design(weights = Weight,
                   strata = Stratum,
                   ids = VarUnit,
                   nest = TRUE)
```

1. How many women have a graduate degree? Hint: the variables `Gender` and `Education` will be useful.

```
#Option 1:
femgd <- anes_des |>
  filter(Gender=="Female", Education=="Graduate") |>
  survey_count(name="n")
#Option 2:
femgd <- anes_des |>
  filter(Gender=="Female", Education=="Graduate") |>
  summarize(
    N=survey_total(),  .groups="drop"
  )
femgd
```

```
# A tibble: 1 x 2
          N    N_se
      <dbl>   <dbl>
1 15108636. 839898.
```

There are 15,108,636 women with a graduate degree.

2. What percentage of people identify as "Strong democrat"? Hint: The variable `PartyID` indicates what party people identify with.

```
(psd <- anes_des |>
   group_by(PartyID) |>
   summarize(
     p=survey_mean()
   ) |>
   filter(PartyID=="Strong democrat"))
```

```
# A tibble: 1 x 3
  PartyID             p    p_se
  <fct>           <dbl>   <dbl>
1 Strong democrat 0.219 0.00646
```

21.9% of people identify as a strong democrat.

3. What percentage of people who voted in the 2020 election identify as "Strong republican"? Hint: The variable `VotedPres2020` indicates whether someone voted in 2020.

```
(psr <- anes_des |>
   filter(VotedPres2020=="Yes") |>
   group_by(PartyID) |>
   summarize(
     p=survey_mean()
   ) |>
   filter(PartyID=="Strong republican"))
```

```
# A tibble: 1 x 3
  PartyID               p    p_se
  <fct>             <dbl>   <dbl>
1 Strong republican 0.224 0.00790
```

22.4% of people identify as a strong republican among those who voted in 2020.

4. What percentage of people voted in both the 2016 election and in the 2020 election? Include the logit confidence interval. Hint: The variable `VotedPres2016` indicates whether someone voted in 2016.

```
(pvb <- anes_des |>
   filter(!is.na(VotedPres2016), !is.na(VotedPres2020)) |>
   group_by(interact(VotedPres2016, VotedPres2020)) |>
   summarize(
     p=survey_prop(var="ci", method="logit"),
   ) |>
   filter(VotedPres2016=="Yes", VotedPres2020=="Yes"))
```

```
When `proportion` is unspecified, `survey_prop()` now defaults to `proportion = TRUE`.
i This should improve confidence interval coverage.
This message is displayed once per session.
```

```
# A tibble: 1 x 5
  VotedPres2016 VotedPres2020     p p_low p_upp
  <fct>         <fct>         <dbl> <dbl> <dbl>
1 Yes           Yes           0.626 0.607 0.644
```

62.6% (60.7-64.4%) voted in both the 2016 and 2020 elections.

5. What is the design effect for the proportion of people who voted early? Hint: The variable `EarlyVote2020` indicates whether someone voted early in 2020.

```
(pdeff <- anes_des |>
   filter(!is.na(EarlyVote2020)) |>
   group_by(EarlyVote2020) |>
   summarize(
     p=survey_mean(deff=TRUE)
   ) |>
   filter(EarlyVote2020=="Yes"))
```

```
# A tibble: 1 x 4
  EarlyVote2020      p    p_se p_deff
  <fct>          <dbl>   <dbl>  <dbl>
1 Yes           0.0525 0.00420   2.27
```

The design effect is 2.27.

6. Were people who lean democrat more likely to vote early in the 2020 election? Use a logistic regression.

```
anes_des |>
  filter(!is.na(PartyID), !is.na(EarlyVote2020)) |>
  group_by(PartyID, EarlyVote2020) |>
  summarise(
    p=survey_mean(),
    .groups="drop"
  ) |>
  filter(EarlyVote2020=="Yes")
```

```
# A tibble: 7 x 4
  PartyID                   EarlyVote2020      p    p_se
  <fct>                     <fct>          <dbl>   <dbl>
1 Strong democrat           Yes           0.0807 0.0101
2 Not very strong democrat  Yes           0.0368 0.00802
3 Independent-democrat      Yes           0.0549 0.00991
4 Independent               Yes           0.0485 0.0131
5 Independent-republican    Yes           0.0352 0.00806
6 Not very strong republican Yes          0.0210 0.00529
7 Strong republican         Yes           0.0502 0.00852
```

```
(pid_vote <- anes_des |>
    svyglm(design=_,
           formula=(EarlyVote2020=="Yes")~PartyID,
           family=quasibinomial(),
           na.action=na.omit))
```

```
Stratified 1 - level Cluster Sampling design (with replacement)
With (101) clusters.
Called via srvyr
Sampling variables:
  - ids: VarUnit
  - strata: Stratum
  - weights: Weight

Call:  svyglm(formula = (EarlyVote2020 == "Yes") ~ PartyID, design = anes_des,
    family = quasibinomial(), na.action = na.omit)

Coefficients:
```

```
                     (Intercept)    PartyIDNot very strong democrat
                         -2.4329                            -0.8332
      PartyIDIndependent-democrat                PartyIDIndependent
                         -0.4138                            -0.5440
    PartyIDIndependent-republican  PartyIDNot very strong republican
                         -0.8787                            -1.4115
         PartyIDStrong republican
                         -0.5065

Degrees of Freedom: 6389 Total (i.e. Null);  45 Residual
  (1063 observations deleted due to missingness)
Null Deviance:      2343
Residual Deviance: 2300     AIC: NA
```

```
summary(pid_vote)
```

```
Call:
svyglm(formula = (EarlyVote2020 == "Yes") ~ PartyID, design = anes_des,
    family = quasibinomial(), na.action = na.omit)

Survey design:
Called via srvyr

Coefficients:
                                 Estimate Std. Error t value Pr(>|t|)
(Intercept)                       -2.4329     0.1360 -17.885  < 2e-16 ***
PartyIDNot very strong democrat   -0.8332     0.2498  -3.336  0.00171 **
PartyIDIndependent-democrat       -0.4138     0.2501  -1.655  0.10496
PartyIDIndependent                -0.5440     0.3192  -1.704  0.09525 .
PartyIDIndependent-republican     -0.8787     0.2599  -3.381  0.00150 **
PartyIDNot very strong republican -1.4115     0.2845  -4.962 1.04e-05 ***
PartyIDStrong republican          -0.5065     0.2090  -2.423  0.01946 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasibinomial family taken to be 0.8978931)

Number of Fisher Scoring iterations: 6
```

Strong Democrats are more likely to vote early.

Bonus

1. What percentage of people lean republican? These are individuals that are strong republicans, not very strong republicans and are independent-republicans. Include an appropriate confidence interval. Hint: to get the correct confidence interval, create a new variable BEFORE calculating the estimate.

```
#Solution 1: Using forcats package
anes_des |>
  filter(!is.na(PartyID)) |>
  mutate(PartyID3=fct_collapse(PartyID,
                               LeanDem=c("Strong democrat",
                                         "Not very strong democrat",
                                         "Independent-democrat"),
                               LeanRep=c("Strong republican",
                                         "Not very strong republican",
                                         "Independent-republican"),
                               other_level="Other")) |>
  group_by(PartyID3) |>
  summarize(p=survey_prop(vartype="ci", proportion = TRUE))
```

```
# A tibble: 3 x 4
  PartyID3     p p_low p_upp
  <fct>    <dbl> <dbl> <dbl>
1 LeanDem  0.448 0.430 0.465
2 LeanRep  0.414 0.397 0.431
3 Other    0.139 0.127 0.151
```

```
#Solution 2: Using case_when
anes_des |>
  filter(!is.na(PartyID)) |>
  mutate(PartyID3=case_when(PartyID %in% c("Strong democrat",
                                           "Not very strong democrat",
                                           "Independent-democrat")~"LeanDem",
                            PartyID %in% c("Strong republican",
                                           "Not very strong republican",
                                           "Independent-republican")~"LeanRep",
                            TRUE~"Other")) |>
  group_by(PartyID3) |>
  summarize(p=survey_prop(vartype="ci", proportion = TRUE))
```

```
# A tibble: 3 x 4
  PartyID3     p p_low p_upp
  <chr>    <dbl> <dbl> <dbl>
1 LeanDem  0.448 0.430 0.465
2 LeanRep  0.414 0.397 0.431
3 Other    0.139 0.127 0.151
```